Fast computation of magnetostatic fields by Non-uniform Fast Fourier Transforms

Evaggelos Kritsikis, Jean-Christophe Toussaint, Olivier Fruchart, Helga Szambolics, L. D. Buda-Prejbeanu

Fast computation of magnetostatic fields by Non-uniform Fast Fourier Transforms

Evaggelos Kritsikis,[1,*] Jean-Christophe Toussaint,[2,3] and Olivier Fruchart[2]

[1]*Spintec, CNRS/CEA/UJF/INP Grenoble, 38054 Grenoble cedex 9, France*
[2]*Institut Néel, CNRS et Université Joseph Fourier, BP 166, F-38042 Grenoble Cedex 9, France*
[3]*INP Grenoble, 38031 Grenoble cedex 1, France*

(Dated: 15 juillet 2008)

The bottleneck of micromagnetic simulations is the computation of the long-ranged magnetostatic fields. This can be tackled on regular N-node grids with Fast Fourier Transforms in time $N \log N$, whereas the geometrically more versatile finite element methods (FEM) are bounded to $N^{4/3}$ in the best case. We report the implementation of a Non-uniform Fast Fourier Transform algorithm which brings a $N \log N$ convergence to FEM, with no loss of accuracy in the results.

The power of computers steadily increases over the years while the size of devices used in fundamental science or technology is shrinking. Today we have reached a cross-over where numerical simulations are capable of describing in detail the physics of nanodevices, for which they thus play a leading role in their understanding and designing. In numerical micromagnetics for spin electronics, a bottleneck is the computation of the magnetostatic interactions which by nature are long-ranged. These interactions can be expressed in terms of either magnetostatic field $\mathbf{H}$ or scalar pseudo-potential ϕ such that $\mathbf{H} = -\nabla\phi$. The latter is convenient since it boils the problem down to a single scalar unknown. To deal with magnetostatic interactions, essentially two distinct approaches have been implemented, depending on the type of mesh used :

1. for Finite Difference (ie. translation invariant) meshes, a Green approach with Fast Fourier Transforms, called FD-FFT. Computation time is moderate ($N \log N$ with N the number of nodes), but the curved boundaries that occur often in experimental devices are not ideally described ;

2. for Finite Element (much more general) meshes, a Finite Element Method coupled to a Boundary Element Method, called FEM-BEM. This can faithfully describe curved boundaries but computation time is higher, at least $N^{3/2}$ in 2D (resp. $N^{4/3}$ in 3D).

In this Letter we report the implementation of a new magnetostatic code which combines the advantages of both cited approaches : it uses a FEM mesh thus describing curved boundaries as well as FEM-BEM, albeit with computation time $N \log N$. It is based on a algorithm reported recently for computing non-periodic Fast-Fourier Transforms (NFFT)[1]. Our code, called FEM-NFFT, proves to be significantly faster than FEM-BEM with no loss of precision. As a first step the implementation was done for a 2D geometry (which pertains to 3D systems with one direction of translational invariance, i.e. cylinder-like) for the proof of concept. The gain is expected to be even greater in more realistic 3D calculations. This demonstrates the potential of FEM-NFFT for micromagnetism and thus spin-electronic devices.

Let us recall the principle and features of FD-FFT and FEM-BEM before presenting our approach and results. We consider a system Ω with boundary $\partial\Omega$, displaying a known magnetization distribution $\mathbf{M}(\mathbf{r})$.

Finite Difference micromagnetic codes use a translation invariant grid. On such a grid, FFTs can be used to compute convolutions in time $N \log N$. This motivates a Green approach for magnetostatics : ϕ is calculated as a convolution of the Green function $\mathsf{G} = -(1/2\pi)\log\mathbf{r}$ in 2D [resp. $\mathsf{G} = 1/(4\pi\mathbf{r})$ in 3D] with the magnetic charges, volumic $\rho = -\nabla\cdot\mathbf{M}$ and surfacic $\sigma = \mathbf{M}\cdot\mathbf{n}$:

$$\phi(\mathbf{r}) = \int_\Omega \rho(\mathbf{r}')\cdot\mathsf{G}(\mathbf{r}-\mathbf{r}')\,\mathrm{d}\mathbf{r}' + \int_{\partial\Omega} \sigma(\mathbf{r}')\cdot\mathsf{G}(\mathbf{r}-\mathbf{r}')\,\mathrm{d}\mathbf{r}' \quad (1)$$

The $N \log N$ speed explains the wide and lasting use of FD in micromagnetic simulations[2]. However, most devices have curved boundaries, either by design or as a result of experimental imperfections. Simulating these cases with FD requires the use of saw-tooth boundaries to describe the magnetic material. This geometrical approximation may induce inadequate descriptions[3].

Finite Element micromagnetic codes use, on the contrary, complex-shaped meshes with triangles (resp. tetrahedrons) as 2D (resp. 3D) unit cells. They consequently suffer much less from the above-mentioned limitations. However, without translational invariance of the mesh, FFTs are not available. Bearing in mind that direct summation of Eq.(1) on a FEM mesh, called FEM-direct, would cost N^2 time, one understands why the Green approach is thought incompatible with FEM codes.

To deal with magnetostatics, FEM codes thus go back to the Poisson equation $-\Delta\phi = \rho$, with the usual regularity condition that the field should decay at infinity. Applying FEM, the equations are translated into a linear system, which is solved by standard iterative methods[4]

*Electronic address: evaggelos.kritsikis@grenoble.cnrs.fr

in time $N^{3/2}$ in 2D (resp. $N^{4/3}$ in 3D) Not only is this asymptotically slower than the $N \log N$ time required for FD-FFT, but N here takes higher values, because the mesh must extend well beyond Ω in order to tackle the regularity condition at infinity. This induces an additional slowdown, and also creates finite-size artifacts.

To avoid meshing outside Ω, the main approach consists in coupling FEM with a Boundary Element Method, resulting in the so-called FEM-BEM[5]. The asymptotic complexity of the Poisson solver is unchanged but the BEM step introduces another time limitation N_∂^2 where N_∂ is the number of boundary nodes. In the most favorable case consisting of compact systems $N_\partial \approx N^{1/2}$ in 2D (resp. $N_\partial \approx N^{2/3}$ in 3D). However for flat geometries, of particular relevance to applications, $N_\partial \approx N$, in which case the time limitation for FEM-BEM may be pretty severe.

Our innovation is to revert, within the FEM framework, to a Green approach. The convolution (1) is discretized in a way typical for FEM, and computed using a fairly recent mathematical method called NFFT (Non-uniform Fast Fourier Transform). NFFTs allows one to compute discrete convolutions in time $N \log N$ without the equispaced data requirement of FFTs.

More in details, we seek ϕ at the nodes $(\mathbf{r}_i)$, $_{i=1\ldots N}$ of the mesh. A linear interpolation inside each element, of known magnetization values $\mathbf{M}(\mathbf{r}_i)$, is used to evaluate charges ρ and σ at points $\mathbf{r}_j$, $_{j=1\ldots M}$ defined as the quadrature points for the integrals in (1)[6].

Consequently, (1) is rewritten the following way :

$$\phi(\mathbf{r}_i) = \sum_{j=1}^{M} \rho_j \, \mathsf{G}(\mathbf{r}_i - \mathbf{r}_j) \, \omega_j \, \det J(r_j) + \sum_{j=1}^{M} \sigma_j \, \mathsf{G}(\mathbf{r}_i - \mathbf{r}_j) \, \omega_j \, \det J(r_j) \quad (2)$$

where

- $\rho_j = \rho(r_j)$ if r_j is in the interior of Ω, 0 otherwise,
- $\sigma_j = \sigma(r_j)$ if r_j is on $\partial\Omega$, 0 otherwise,
- ω_j is the weight of r_j in the quadrature scheme,
- $J(r_j)$ is the Jacobian of the affine transformation mapping the unit element on that containing r_j.

We then use NFFTs to compute (2). Although the seminal paper[7] dates back to 1993, the NFFT method remains little-known even in the mathematical community. A presentation of the method can be found Ref.[8]. Here we sketch the basic strategy and show that computation time does not exceed $N \log N$, ie. that of the classical FFT.

Let us look at our non equispaced data as a sum of Dirac functions. The goal is to find the spectrum of this data function. The idea is to *convey initial information over a regular grid*, so that an FFT can be used. We therefore choose a regular grid X and let the data diffuse to the X nodes through convolution with a Gaussian function (or more generally a smooth localized function). The Gaussian is localized in space, so we can consider that each piece of data diffuses only to a fixed number of the nearest X nodes. Computation time of this diffusion step is thus proportional to N.

The question arises how to choose the period of the X grid. In our case, the answer depends on how smooth the Green function G is. A necessary preliminary step before executing the NFFT is therefore to smooth G around the origin ; the price to pay is an afterwards correction in the smoothing zone. It can be shown that a grid of size $p^2 N$ in 2D (resp. $p^3 N$ in 3D) is convenient, where p is the degree of smoothness chosen for G.

We then perform, according to the initial idea, a FFT on the X grid, in time proportional to $N \log N$. Based on the convolution theorem, what we get is the Fourier coefficients of the data function *multiplied* by those of the Gaussian. Therefore, we finally divide these numbers by the Fourier coefficients of the Gaussian to get the desired spectrum. The number of divisions is proportional to N. As a whole, the NFFT is expected to behave asymptotically like $N \log N$, as all extra steps behave like N.

We implemented FEM-NFFT to 2D test cases where an analytical solution ϕ_a is available, so that errors can be readily estimated. Interpolation and quadrature routines are written in C++ and the NFFT package used[9] is in C99. For G we have chosen a smoothness degree of 2. On each test case, we provide computation times and error estimates for FEM-direct, the classic FEM-BEM and our NFFT-based method. The computed error is the normalized root mean square $(M_s L)^{-1}(\sum_{i=1}^{N} |\phi(\mathbf{r}_i) - \phi_\mathrm{a}(\mathbf{r}_i)|^2/N)^{1/2}$, where M_s is the saturation magnetization and L the system diameter set at unity, . Computations are done on an Intel P4 2GHz with 1GB RAM running Fedora 5.

The first test case is a disk uniformly magnetized along the x-axis (a cylinder in 3D space). The analytical solution is $\phi(x) = M_s \, x/2$. The second case is the so-called magic cylinder, a circular annulus of radii R_1, R_2 where the angle between magnetization and the x-axis equals twice the polar angle. The name stems from the uniform magnetic field thus induced in the inner region. The analytical solution inside the annulus is $\phi(r, \theta) = M_s \, r \cos\theta \log(r/R_2)$ in polar coordinates.

Tables 1 and 2 display the numerical results for the two cases, respectively. It can be readily seen that FEM-NFFT provides results very similar to FEM-direct. The error induced by the NFFTs is thus negligible. Compared to FEM-BEM, errors are comparable for non-uniformly magnetized systems (see Table 2) whereas for uniform distributions, Green approaches, to which FEM-NFFT belong, are more accurate by one order of magnitude (see Table 1). This is because they can treate apart volu-

mic and surfacic charge contributions. Concerning computational time, FEM-direct is as expected the quickest, gaining a factor around 5 over FEM-BEM for the finer meshes. Nodes required in 3D cases of interest commonly count up to 10^5, around which number the time advantage of FEM-NFFT over classical methods such as FEM-BEM is expected to reach one order of magnitude.

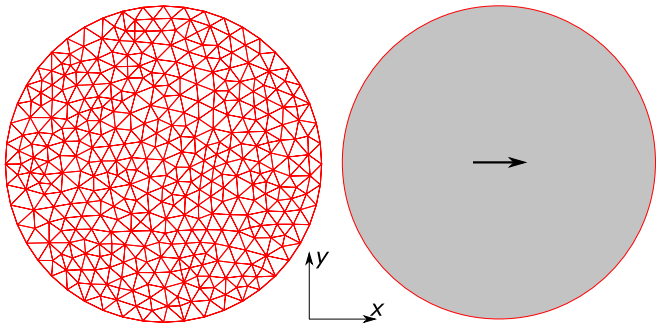

FIG. 1 – Test case 1 : the uniformly magnetized disk. Mesh used for $N = 400$ (left) and magnetization distribution (right).

TAB. I – Test case 1 : error (in ppm) and computation time (in seconds) of FEM-direct, FEM-BEM and FEM-NFFT for different mesh sizes.

N	error			time		
	FEM-direct	FEM-BEM	FEM-NFFT	FEM-direct	FEM-BEM	FEM NFFT
400	93.6	162	90.3	0.401	0.070	0.089
1572	17.3	52.8	16.8	7.71	0.390	0.385
9489	2.16	25.8	2.09	233	4.75	2.18
37938	−−	23.2	0.650	3720*	40.1	9.81

*: estimated.

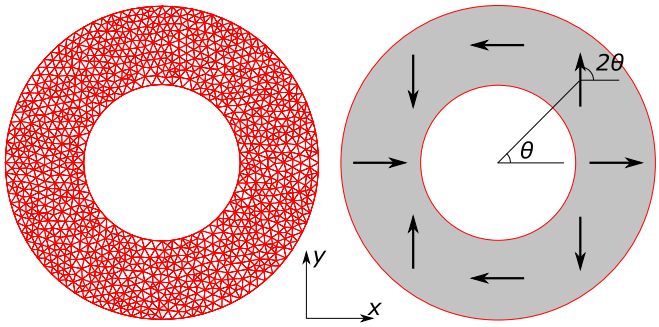

FIG. 2 – Test case 2 : the magic cylinder. Mesh used for $N = 1192$ (left) and magnetization distribution (right).

To conclude, we have successfully implemented a Non-uniform Fast Fourier Transform (NFFT) algorithm to compute magnetostatic fields for micromagnetic simulations based on Finite Element methods (FEM). The new approach, called FEM-NFFT, combines the advantages previously found separately in Finite Difference methods (computation time scaling like $N \log N$) and FEM (faithful description of curved boundaries). Thus FEM-NFFT promises a leap in the attractiveness of micromagnetic simulations of spin electronic devices.

TAB. II – Test case 2 : error (in ppm) and computation time (in seconds) of FEM-direct, FEM-BEM and FEM-NFFT for different mesh sizes.

N	error			time		
	FEM-direct	FEM-BEM	FEM-NFFT	FEM-direct	FEM-BEM	FEM NFFT
1192	174	172	174	3.58	0.290	0.331
2091	95.8	94.2	95.8	14.0	0.640	0.498
8214	24.7	32.1	24.8	211	4.59	2.02
22683	−−	21.5	9.05	1600*	34.0	6.62

*: estimated.

[1] G. S. D. Potts and M. Tasche, *Fast Fourier transforms for nonequispaced data : A tutorial* (Birkhäuser, Boston, 2001), pp. 247–270.

[2] B. Kevorkian, Ph.D. thesis, Université Joseph Fourier, Grenoble (1998).

[3] Z. G. C. J. García-Cervera and E. Weinan, J. Comput.

Phys. **184** (2003).

[4] D. Braess, *Finite Elements* (Cambridge University Press, Cambridge, 2001), 2nd ed.

[5] D. Fredkin and T. Koehler, IEEE Transactions on Magnetics **26** (1990).

[6] In singular zones where the quadrature scheme becomes not precise enough, it is possible to combine it with semi-analytical terms, the details of which will come in a forthcoming paper.

[7] A. Dutt and V. Rokhlin, SIAM J. Sci. Stat. Comput. **14**, 1368 (1993).

[8] L. Greengard and J.-Y. Lee, SIAM Rev. **46**, 443 (2004).

[9] S. Kunis and D. Potts, *NFFT, Softwarepackage, C subroutine library* (2002-06).